Minimal dominating sets in graph classes: combinatorial bounds and enumeration

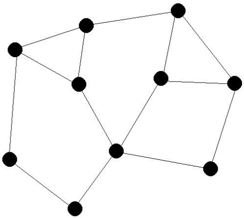
J.-F. Couturier[1] P. Heggernes[2]
D. Kratsch[1] P. van 't Hof[2]

[1]LITA
Université Paul Verlaine - Metz
Metz, France

[2]Department of Informatics
University of Bergen
Bergen, Norway

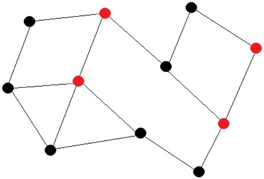
SOFSEM 2012

$G = (V, E)$ simple undirected graph.
V its vertex set.
E its edge set.



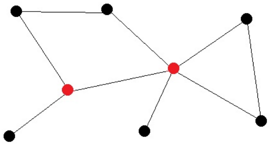
A set D is a *dominating set* of the graph $G = (V, E)$, if $\forall v \in V$:

- either $v \in D$
- or $\exists x \in D$ such that $vx \in E$



Minimum dominating set

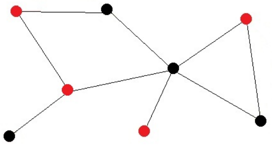
- Input : graph $G = (V, E)$
- Output : minimum cardinality of a dominating set D of G



This problem is NP-complete.
The best known exact algorithm runs in $O^*(1.4957^n)$ [J. van Rooij].

A set D is a *minimal dominating set* of the graph $G = (V, E)$ if D is a dominating set, and $\forall x \in D$

- either x has no neighbour in D
- or $\exists$ a neighbour $y \in V \setminus D$ of x such that y has no neighbour in $D \setminus \{x\}$. y is called a *private neighbour* of x.



Inclusion minimal dominating set

- Input : graph $G = (V, E)$
- Output : an inclusion minimal dominating set D of G.

This problem is polynomial time solvable !

Inclusion minimal dominating set

- Input : graph $G = (V, E)$
- Output : an inclusion minimal dominating set D of G.

This problem is polynomial time solvable !

What if one minimal dominating set is not enough ?

Enumerating all minimal dominating sets

- Input : graph $G = (V, E)$
- Output : all minimal dominating sets of G.

Enumerating all minimal dominating sets allows immediate solution of corresponding NP-hard optimisation and counting problems.

Combinatorial Question

How many minimal dominating sets may a graph on n vertices have ? Not more than 2^n but ...

What is the maximum number of minimal dominating sets in a graph on n vertices ?

An upper bound was given in 2008 by F. V. Fomin, F. Grandoni, A. V. Pyatkin, and A. A. Stepanov.

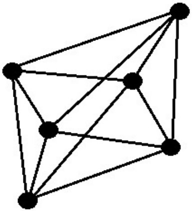
The number of minimal dominating sets in a graph on n vertices is at most 1.7159^n.

What is the maximum number of minimal dominating sets in a graph on n vertices ?

Fomin et al. also give a lower bound.

There is a graph on n vertices with $15^{n/6}$ minimal dominating sets.

This gives a lower bound of 1.5704^n for the maximum number of minimal dominating sets.

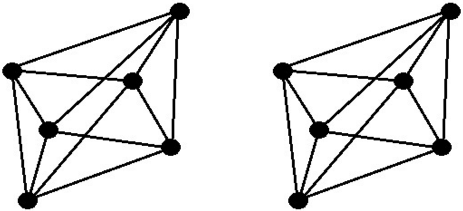


What is the maximum number of minimal dominating sets in a graph on n vertices ?

Fomin et al. also give a lower bound.

There is a graph on n vertices with $15^{n/6}$ minimal dominating sets.

This gives a lower bound of 1.5704^n for the maximum number of minimal dominating sets.

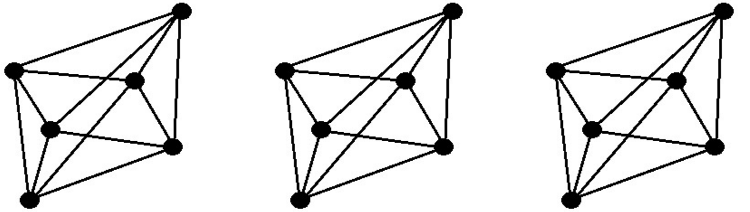


What is the maximum number of minimal dominating sets in a graph on n vertices ?

Fomin et al. also give a lower bound.

There is a graph on n vertices with $15^{n/6}$ minimal dominating sets.

This gives a lower bound of 1.5704^n for the maximum number of minimal dominating sets.



Not tight !

There is a huge gap between the lower bound 1.5704^n and the upper bound 1.7159^n.

No improvements have been achieved until today.

Graph classes

Our work is dealing with some well-known graph classes. The goal is to find corresponding lower and upper bounds.

Graph classes

Our work is dealing with some well-known graph classes. The goal is to find corresponding lower and upper bounds.

Why graph classes ?

We attempt to exploit the particular structure of various graph classes to achieve better bounds, preferably even *matching* upper and lower bounds.

Reminder : general case

We have alredy mentioned

Lower bound	Upper bound
1.5704^n	1.7159^n

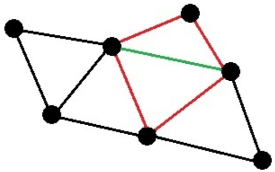
In the following we summarize our results :

Some graph classes and the corresponding bounds :

Graph Class	Lower Bound	Upper Bound
chordal	1.4422^n	1.6181^n
split	1.4422^n	1.4656^n
proper interval	1.4422^n	1.4656^n
trivially perfect*	1.4422^n	1.4423^n

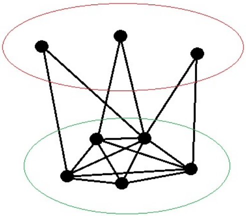
A graph is chordal if every cycle of length at least 4 has a chord.

Graph Class	Lower Bound	Upper Bound
chordal	1.4422^n	1.6181^n
split	1.4422^n	1.4656^n
proper interval	1.4422^n	1.4656^n
trivially perfect*	1.4422^n	1.4423^n



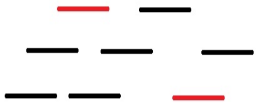
A graph is a split graph if its vertex set can be partitioned in an independent set and a clique.

Graph Class	Lower Bound	Upper Bound
chordal	1.4422^n	1.6181^n
split	1.4422^n	1.4656^n
proper interval	1.4422^n	1.4656^n
trivially perfect*	1.4422^n	1.4423^n



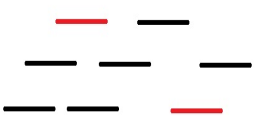
A proper interval graph is an interval graph having an intersection model in which no interval properly contains another one.

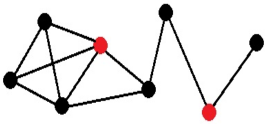
Graph Class	Lower Bound	Upper Bound
chordal	1.4422^n	1.6181^n
split	1.4422^n	1.4656^n
proper interval	1.4422^n	1.4656^n
trivially perfect*	1.4422^n	1.4423^n



A proper interval graph is an interval graph having an intersection model in which no interval properly contains another one.

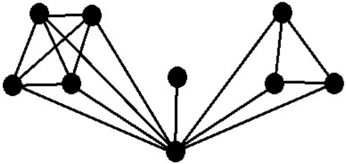
Graph Class	Lower Bound	Upper Bound
chordal	1.4422^n	1.6181^n
split	1.4422^n	1.4656^n
proper interval	1.4422^n	1.4656^n
trivially perfect*	1.4422^n	1.4423^n





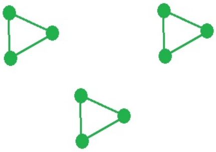
A graph is trivially perfect if it has neither P_4 nor C_4 as induced subgraph.

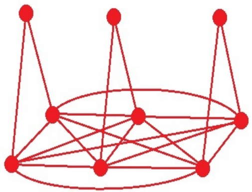
Graph Class	Lower Bound	Upper Bound
chordal	1.4422^n	1.6181^n
split	1.4422^n	1.4656^n
proper interval	1.4422^n	1.4656^n
trivially perfect*	1.4422^n	1.4423^n



In this table, all graph classes have the same lower bound.
The 1.4422^n lower bound is achieved by two types of graphs on n vertices, both having $3^{\frac{n}{3}}$ minimal dominating sets.

Graph Class	Lower Bound	Upper Bound
* chordal *	1.4422^n	1.6181^n
split *	1.4422^n	1.4656^n
* proper interval	1.4422^n	1.4656^n
* trivially perfect*	1.4422^n	1.4423^n





More lower and upper bounds on the maximum number of minimal dominating sets in a graph on n vertices in certain graph classes :

Graph Class	Lower Bound	Upper Bound
cobipartite	1.3195^n	1.5875^n
cograph*	1.5704^n	1.5705^n
threshold*	$\omega(G)$	$\omega(G)$
chain*	$\lfloor n/2 \rfloor + m$	$\lfloor n/2 \rfloor + m$
forest	1.4142^n	1.4656^n

... can we see an algorithm now ?

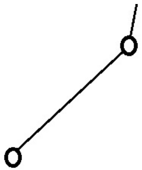
Why an algorithm ?

The execution of a branching algorithm can be represented by a search tree.

If the algorithm enumerate all solution, when the execution is finished, all solution is contain in a leaf of the search tree.

Why an algorithm ?

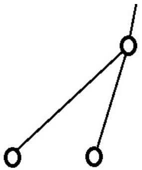
The execution of a branching algorithm can be represented by a search tree.



If the algorithm enumerate all solution, when the execution is finished, all solution is contain in a leaf of the search tree.

Why an algorithm ?

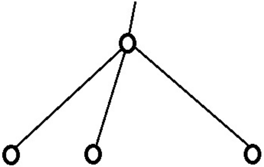
The execution of a branching algorithm can be represented by a search tree.



If the algorithm enumerate all solution, when the execution is finished, all solution is contain in a leaf of the search tree.

Why an algorithm ?

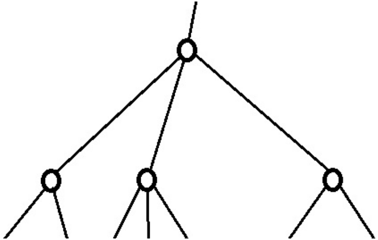
The execution of a branching algorithm can be represented by a search tree.



If the algorithm enumerate all solution, when the execution is finished, all solution is contain in a leaf of the search tree.

Why an algorithm ?

The execution of a branching algorithm can be represented by a search tree.



If the algorithm enumerate all solution, when the execution is finished, all solution is contain in a leaf of the search tree.

The number of leaf in the search tree is an upper bound !

If we can bound the number of leaves in the search tree, we bound at the same time the number of solutions of the problem !
And bound the number of leaves in the search tree is exactly what an estimation of execution time does.

The number of leaf in the search tree is an upper bound !

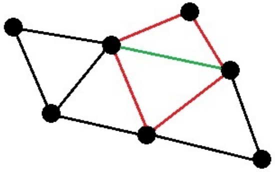
If we can bound the number of leaves in the search tree, we bound at the same time the number of solutions of the problem !
And bound the number of leaves in the search tree is exactly what an estimation of execution time does.

Be careful, it is an upper bound !

Every solution is in a leaf, but every leaf does not have a solution.

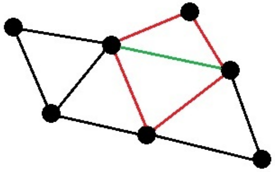
Chordal graphs

A graph is chordal if every cycle of length at least 4 has a chord.



Chordal graphs

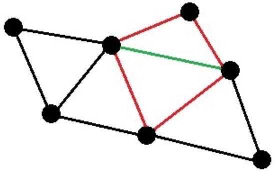
A graph is chordal if every cycle of length at least 4 has a chord.



Every chordal graph has a simplicial vertex.

Chordal graphs

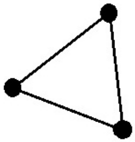
A graph is chordal if every cycle of length at least 4 has a chord.



Every chordal graph has a simplicial vertex.
A vertex x is *simplicial* if its neighbourhood $N(x)$ is a clique.

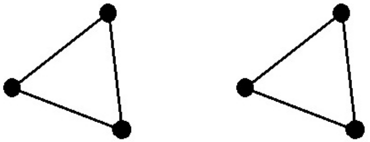
A lower bound of 1.4422^n

Take a disjoint union of $n = 3t$ triangles.
This chordal graph has $3^{\frac{n}{3}}$ minimal dominating sets.



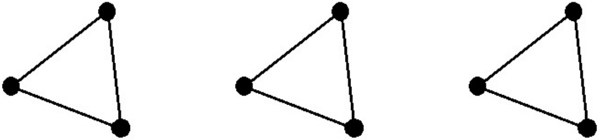
A lower bound of 1.4422^n

Take a disjoint union of $n = 3t$ triangles.
This chordal graph has $3^{\frac{n}{3}}$ minimal dominating sets.



A lower bound of 1.4422^n

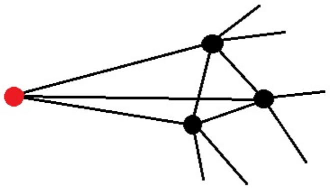
Take a disjoint union of $n = 3t$ triangles.
This chordal graph has $3^{\frac{n}{3}}$ minimal dominating sets.



Our algorithm to enumerate all minimal dominating sets of a chordal graph always chooses a simplicial vertex x to branch on. There are three different types of branchings.

Case 1 : x is already dominated.

- $x \in D$. Since x is simplicial and needs a private neighbour in $N(x)$, we can delete x and all its neighbours.
- $x \notin D$. Since it is already dominated, it is safe to delete x.

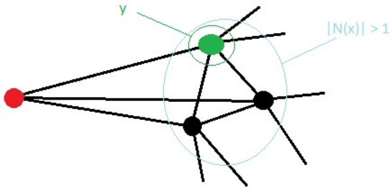


Thus the branching vector is $(2, 1)$.

Case 2 : x is not already dominated and $|N(x)| \geq 2$

Let y be a neigbour of x.

- $y \in D$. Since x is simplicial, all neighbours of x are dominated by y. We delete x and y.
- $y \notin D$. Since $y \in N(x)$, any vertex we select later to dominate x will also dominate y. Thus we can delete y.

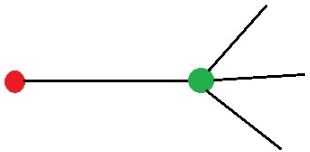


Thus the branching vector is $(2, 1)$.

Case 3 : x is not already dominated and $|N(x)| \leq 1$.

Let y be the neighbour of x.

- $x \in D$. Since y is the private neighbour of x, we can delete x and y.
- $x \notin D$. The only way to dominate x is to take y into D. Hence $y \in D$ and we can delete x and y.



Thus the branching vector is $(2, 2)$.

Running time of algorithm

Our three branching rules have branching vectors $(2, 1)$, $(2, 1)$ and $(2, 2)$.
The worst case is due to the branching vector $(2, 1)$. This implies that the enumeration algorithm has a running time of $O^*(1.6181^n)$.

Upper bound

This also implies an upper bound of $O^*(1.6181^n)$ for the number of minimal dominating sets in a chordal graph on n vertices.

Lower bound

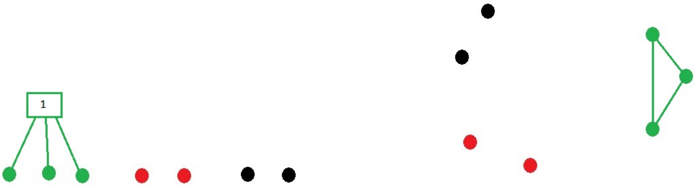
Recall that the lower bound for chordal graphs is 1.4422^n.

Cographs

A graph G is a *cograph* if it can be constructed from isolated vertices by the operations *disjoint union* and *join*.
This construction can be represented by a *cotree*.

Cographs

A graph G is a *cograph* if it can be constructed from isolated vertices by the operations *disjoint union* and *join*.
This construction can be represented by a *cotree*.

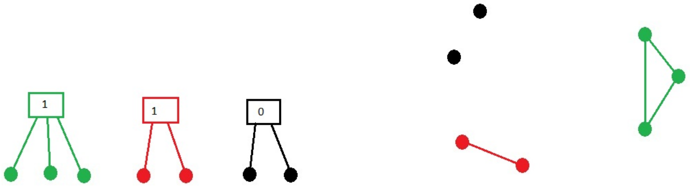


Cographs

A graph G is a *cograph* if it can be constructed from isolated vertices by the operations *disjoint union* and *join*.
This construction can be represented by a *cotree*.

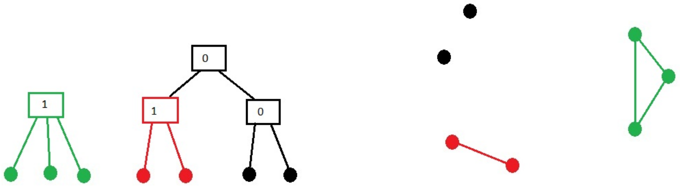
Cographs

A graph G is a *cograph* if it can be constructed from isolated vertices by the operations *disjoint union* and *join*.
This construction can be represented by a *cotree*.



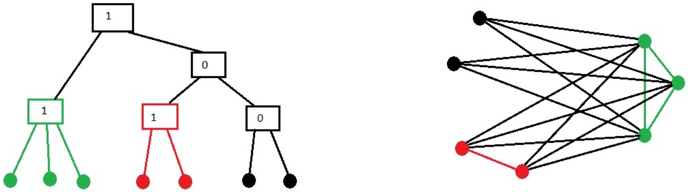
Cographs

A graph G is a *cograph* if it can be constructed from isolated vertices by the operations *disjoint union* and *join*.
This construction can be represented by a *cotree*.



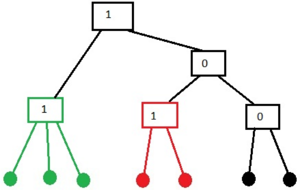
Cographs

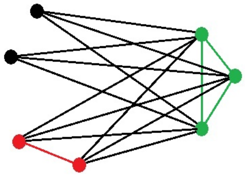
A graph G is a *cograph* if it can be constructed from isolated vertices by the operations *disjoint union* and *join*.
This construction can be represented by a *cotree*.



Cographs

A graph G is a *cograph* if it can be constructed from isolated vertices by the operations *disjoint union* and *join*.
This construction can be represented by a *cotree*.

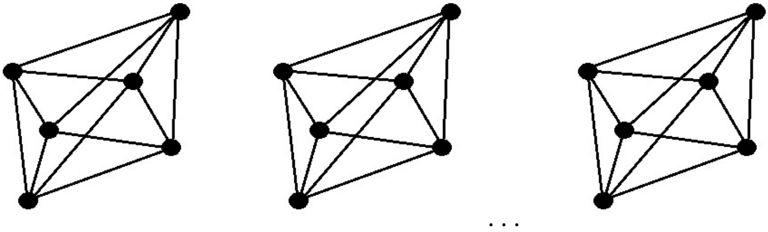




A graph is a cograph iff it has no P_4 as induced subgraph.

Lower bound.

The lower bound graph for the general case is indeed a cograph.



There is a cograph with $15^{\frac{n}{6}}$ minimal dominating sets.

Theorem

Every cograph has at most $15^{\frac{n}{6}}$ minimal dominating sets.

Theorem

Every cograph has at most $15^{\frac{n}{6}}$ minimal dominating sets.

Proof by induction.

It is not difficult to enumerate all the possible cographs with $n \leq 6$ vertices and to verify that each has at most $15^{\frac{n}{6}}$ minimal dominating sets.

Theorem

Every cograph has at most $15^{\frac{n}{6}}$ minimal dominating sets.

Proof by induction.

It is not difficult to enumerate all the possible cographs with $n \leq 6$ vertices and to verify that each has at most $15^{\frac{n}{6}}$ minimal dominating sets.

Assume the theorem is true for all cographs with less than n vertices ...

Let $G = (V, E)$ be a cograph.

Every cograph can be constructed from isolated vertices by disjoint union and by join operation.
Hence G can be partitioned into graphs G_1 with n_1 vertices and G_2 with n_2 vertices such that :

- if G is a disjoint union of G_1 and G_2, then there is no edge between G_1 and G_2.
- if G is a join of G_1 and G_2, then all the edges with one endpoint in G_1 and one in G_2 are present in G.

Note that $n = n_1 + n_2$.

Let $\mu(G)$ be the number of minimal dominating sets in G.

Case 1 : G is a disjoint union of G_1 and G_2.

Since every minimal dominating set D of G is the union of a minimal dominating set D_1 of G_1 and a minimal dominating set D_2 of G_2, we have :

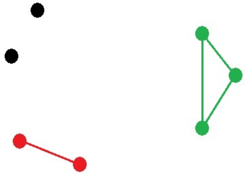
$$\mu(G) = \mu(G_1) \cdot \mu(G_2)$$

Using induction hypothesis for G_1 and G_2, we obtain that the number of minimal dominating sets in G is at most $15^{\frac{n_1}{6}} \cdot 15^{\frac{n_2}{6}} = 15^{\frac{n}{6}}$.

Case 2 : G is a join of G_1 and G_2.

Since for each vertex x_1 of G_1 and for each vertex x_2 of G_2, there is an edge x_1x_2 in G, there are three types of minimal dominating sets of G.

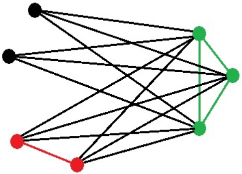
- a minimal dominating set D_1 of G_1,
- a minimal dominating set D_2 of G_2, and
- $\{x_1, x_2\}$ for all vertices x_1 of G_1 and all vertices x_2 of G_2.



Case 2 : G is a join of G_1 and G_2.

Since for each vertex x_1 of G_1 and for each vertex x_2 of G_2, there is an edge x_1x_2 in G, there are three types of minimal dominating sets of G.

- a minimal dominating set D_1 of G_1,
- a minimal dominating set D_2 of G_2, and
- $\{x_1, x_2\}$ for all vertices x_1 of G_1 and all vertices x_2 of G_2.



Case 2 : G is a join of G_1 and G_2.

Consequently :

$$\mu(G) = \mu(G_1) + \mu(G_2) + n_1 \cdot n_2$$

Using induction hypothesis for G_1 and G_2 and the fact that $n \geq 7$, we obtain that the number of minimal dominating sets in G is at most $15^{\frac{n_1}{6}} + 15^{\frac{n_2}{6}} + n_1 \cdot n_2 \leq 15^{\frac{n}{6}}$.

Lower bound matches upper bound

$15^{\frac{n}{6}}$ is a tight upper bound for the maximum number of minimal dominating sets in a cograph on n vertices.

Future work

- Various bounds are not tight. Improving bounds for general graphs might be hard. Improving bounds for some graph classes might be easier.
- Output sensitive approach to enumeration : constructing output polynomial or even polynomial delay algorithms to enumerate all minimal dominating sets.
- Could our enumeration algorithms be used to establish fast exact exponential algorithms solving the NP-hard problems Domatic Number and Connected Dominating Set on split and chordal graphs ?

Thank you !

F. V. Fomin, F. Grandoni, A. V. Pyatkin, and A. A. Stepanov. Combinatorial bounds via measure and conquer : Bounding minimal dominating sets and applications. ACM Trans. Algorithms 5(1) : (2008).